

PCT
 WELTORGANISATION FÜR GEISTIGES EIGENTUM
 Internationales Büro
 INTERNATIONALE ANMELDUNG VERÖFFENTLICHT NACH DEM VERTRAG ÜBER DIE
 INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS (PCT)



(51) Internationale Patentklassifikation 6 : <p style="text-align: center;">G05B 19/05</p>	A1	(11) Internationale Veröffentlichungsnummer: WO 96/33450 (43) Internationales Veröffentlichungsdatum: 24. Oktober 1996 (24.10.96)
(21) Internationales Aktenzeichen: PCT/DE96/00619 (22) Internationales Anmeldedatum: 9. April 1996 (09.04.96) (30) Prioritätsdaten: <div style="display: flex; justify-content: space-between;"> 195 14 471.6 19. April 1995 (19.04.95) DE </div> (71) Anmelder (für alle Bestimmungsstaaten ausser US): SIEMENS AKTIENGESELLSCHAFT [DE/DE]; Wittelsbacherplatz 2, D-80333 München (DE). (72) Erfinder; und (75) Erfinder/Anmelder (nur für US): REHM, Thomas [DE/DE]; Am Erlanger Weg 45, D-91052 Erlangen (DE). WILLERT, Josef [DE/DE]; Auracher Strasse 1, D-91093 Heßdorf (DE).		(81) Bestimmungsstaaten: CN, KR, US, europäisches Patent (AT, BE, CH, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE). Veröffentlicht <i>Mit internationalem Recherchenbericht.</i> <i>Vor Ablauf der für Änderungen der Ansprüche zugelassenen Frist. Veröffentlichung wird wiederholt falls Änderungen eintreffen.</i>

(54) Title: **PROCESS FOR OPERATING A NUMERICAL CONTROL WITH CRITICAL-TIME AND NON-CRITICAL-TIME PROCESSES IN A REAL TIME SYSTEM**

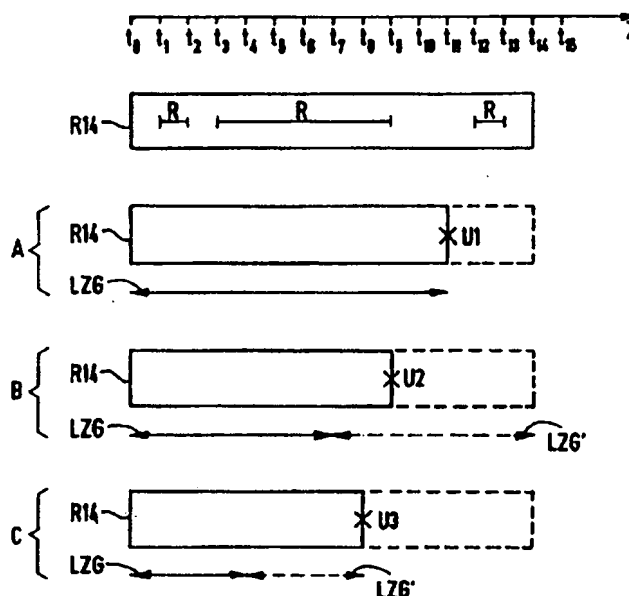
(54) Bezeichnung: **VERFAHREN ZUM BETRIEB EINER NUMERISCHEN STEUERUNG MIT ZEITKRITISCHEN UND NICHT ZEITKRITISCHEN PROZESSEN IN EINEM ECHTZEITSYSTEM**

(57) Abstract

In order, in a real-time system of a digital control, to prevent an individual program sequence (PS2) of a non-critical computing process ("task") from using up a load-dependently disproportionate amount of the total computing time still available, that program sequence (PS2) is classed as a subordinate computing process (R14) with a running time limit (LZF). Program sequences which should preferably not be interrupted are protected against interruption by a lock. The original running time limit (LZG) can be overcome only within predetermined limits.

(57) Zusammenfassung

Um bei einem Echtzeitsystem einer digitalen Steuerung zu verhindern, daß eine einzelne Programmsequenz (PS2) eines nicht zeitkritischen Rechenprozesses ("Task") lastabhängig unverhältnismäßigerweise die gesamte noch zur Verfügung stehende Rechenzeit aufbraucht, wird diese Programmsequenz (PS2) als ein untergeordneter Rechenprozeß (R14) mit einer Laufzeitbegrenzung (LZF) abgesetzt. Vorzugsweise nicht zu unterbrechende Programmschrittfolgen werden dabei über einen Riegel ("Lock") vor Unterbrechung geschützt, wobei die ursprüngliche Laufzeitbegrenzung (LZG) nur innerhalb vorbestimmter Grenzen überzogen werden kann.



LEDIGLICH ZUR INFORMATION

Codes zur Identifizierung von PCT-Vertragsstaaten auf den Kopfbögen der Schriften, die internationale Anmeldungen gemäss dem PCT veröffentlichen.

AM	Armenien	GB	Vereinigtes Königreich	MX	Mexiko
AT	Österreich	GE	Georgien	NE	Niger
AU	Australien	GN	Guinea	NL	Niederlande
BB	Barbados	GR	Griechenland	NO	Norwegen
BE	Belgien	HU	Ungarn	NZ	Neuseeland
BF	Burkina Faso	IE	Irland	PL	Polen
BG	Bulgarien	IT	Italien	PT	Portugal
BJ	Benin	JP	Japan	RO	Rumänien
BR	Brasilien	KE	Kenya	RU	Russische Föderation
BY	Belarus	KG	Kirgisistan	SD	Sudan
CA	Kanada	KP	Demokratische Volksrepublik Korea	SE	Schweden
CF	Zentrale Afrikanische Republik	KR	Republik Korea	SG	Singapur
CG	Kongo	KZ	Kasachstan	SI	Slowenien
CH	Schweiz	LI	Liechtenstein	SK	Slowakei
CI	Côte d'Ivoire	LK	Sri Lanka	SN	Senegal
CM	Kamerun	LR	Liberia	SZ	Swasiland
CN	China	LK	Litauen	TD	Tschad
CS	Tschechoslowakei	LU	Luxemburg	TG	Togo
CZ	Tschechische Republik	LV	Lettland	TJ	Tadschikistan
DE	Deutschland	MC	Monaco	TT	Trinidad und Tobago
DK	Dänemark	MD	Republik Moldau	UA	Ukraine
EE	Estland	MG	Madagaskar	UG	Uganda
ES	Spanien	ML	Mali	US	Vereinigte Staaten von Amerika
FI	Finnland	MN	Mongolei	UZ	Usbekistan
FR	Frankreich	MR	Mauritanien	VN	Vietnam
GA	Gabon	MW	Malawi		

VERFAHREN ZUM BETRIEB EINER NUMERISCHEN STEUERUNG MIT ZEITKRITISCHEN UND NICHT ZEITKRITISCHEN PROZESSEN IN EINEM ECHTZEITSYSTEM.

- 5 Die Erfindung bezieht sich auf ein Verfahren zum Betrieb einer Echtzeitsystem aufweisenden numerischen Steuerung, insbesondere für Werkzeugmaschinen und Roboter, wobei das Echtzeitsystem zeitkritische Rechenprozesse und nicht zeitkritische Rechenprozesse mit je mindestens einer, aus abzu-
- 10 arbeitenden Programmschrittfolgen bestehenden, Programmsequenz aufweist.

Bei einer digitalen Steuerung, insbesondere einer numerischen Steuerung für Werkzeugmaschinen oder Roboter, gibt es zum einen zyklisch wiederkehrende zeitkritische Rechenprozesse.

15 Hierbei handelt es sich beispielsweise um die für die Lage- und Drehzahlregelung notwendige fortlaufende Ausgabe von Sollwerten an die Antriebe der Werkzeugmaschine oder des Roboters. Ebenso fällt hierunter die ständige Erfassung der Lage- und Drehzahlgeberwerte.

20

Andere Aufgaben einer solchen Steuerung, wie z.B. das Aktualisieren der Anzeigeeinheit, das Abfragen der Tastatur der Eingabeeinheit, das Einlesen von künftigen Teileprogrammen, die Aufbereitung von NC-Daten aufgrund der Teileprogramme etc. sind als demgegenüber als nicht zeitkritische Rechenprozesse anzusehen. Rechenprozesse können auch als "Tasks" bezeichnet werden.

25

- 30 Das Echtzeitsystem der Steuerung muß also dafür sorgen, daß die zeitkritischen Rechenprozesse immer sofort abgearbeitet werden. Dazu wird bei Bedarf ein nicht zeitkritischer Rechenprozeß, der gerade abgearbeitet wird, unterbrochen. Die hierzu notwendigen Maßnahmen, z.B. Prozessorinterrupts, sind dem Fachmann bekannt.
- 35

Das Echtzeitsystem muß aber auch dafür sorgen, daß in der noch zur Verfügung stehenden Rechenzeit zwischen der Abarbeitung der zeitkritischen Rechenprozesse die nicht zeitkritischen Rechenprozesse ebenfalls abgearbeitet werden. Bei der Abarbeitung dieser nicht zeitkritischen Rechenprozesse ist ferner zu berücksichtigen, daß alle nicht zeitkritischen Rechenprozesse möglichst zeitgleich nach dem "time sharing" Verfahren abgearbeitet werden.

- 10 Jeder Rechenprozeß besteht aus einer oder mehreren Programmsequenzen, die wiederum aus einer Folge einzelner abzuarbeitender Programmschritte bestehen. Einige dieser Programmsequenzen sind stark lastabhängig. So braucht ein Rechenprozeß, der ein NC-Teileprogramm in den Speicher des Echtzeitsystems
- 15 lädt, für ein großes komplexes Teileprogramm wesentlich mehr Rechenzeit als für ein kleines einfaches Teileprogramm.

Diese Lastabhängigkeit bedeutet, daß die für einzelne Programmsequenzen benötigte Rechenzeit stark schwankt und unter Umständen im Vergleich zu anderen Programmsequenzen unverhältnismäßig groß wird. Es entsteht somit das Problem, daß eine einzelne Programmsequenz die gesamte noch zur Verfügung stehenden Rechenzeit zwischen der Abarbeitung der zeitkritischen Rechenprozesse aufbraucht und daher die anderen Programmsequenzen, z.B. die Bildschirmanzeige, erst einmal nicht weiter abgearbeitet werden. Ein daraus resultierendes Verhalten der Steuerung könnte den Benutzer beunruhigen oder zu Fehlverhalten verleiten.

- 30 Eine bisherige Lösung bestand darin, solche Programmsequenzen in eine Vielzahl von eigenständigen Rechenprozessen aufzuteilen. Dies hatte aber den Nachteil, daß zusätzliche Rechenzeit im Echtzeitsystem für die Verwaltung dieser zusätzlichen Rechenprozesse gebraucht wird; eben auch in
- 35 Fällen, wo die Programmsequenz lastbedingt gerade nicht

unverhältnismäßig groß ist. Ferner sind derartige Programm-Sequenzen nicht immer im Voraus lokalisierbar. Oft werden solche Programmsequenzen nämlich erst bei der Anpassung der Steuerungen an besondere Kundenbedürfnisse ermittelt. Nach-
5 trägliche Änderungen der Rechenprozeßstruktur an die Kundenbedürfnisse wären dann aber äußerst aufwendig. Wenn zudem wirklich nur zur Laufzeit erkennbar ist, ob eine Programmsequenz unverhältnismäßig viel Rechenzeit braucht, müßte eine dynamische Änderung der Rechenprozeßstruktur vorgenommen
10 werden. Dies würde im hohen Maß ebenfalls Rechenzeit erfordern, und wegen der erhöhten Komplexität wären Eingriffe in den laufenden Prozeß auch aus Sicherheitsgründen nicht wünschenswert.

15 Aufgabe der Erfindung ist es, ein Verfahren der eingangs genannten Art so auszubilden, daß stark lastabhängige nicht zeitkritische Programmsequenzen in einfacher Weise daran gehindert werden, die gesamte noch zur Verfügung stehenden Rechenzeit zwischen der Abarbeitung der zeitkritischen
20 Rechenprozesse aufzubreuchen. Dabei gehen die Erfinder im Ansatz von einer Laufzeitbegrenzung solcher Programmsequenzen aus.

Gemäß der Erfindung wird diese Aufgabe dadurch gelöst,
25 - daß zur Laufzeitbegrenzung einer Programmsequenz eines nicht zeitkritischen Rechenprozesses, die Programmsequenz als ein dem Rechenprozeß untergeordneter Rechenprozeß mit einem Laufzeitbegrenzungswert abgesetzt und so abgearbeitet wird,
30 - daß der untergeordnete Rechenprozeß vor Beginn vorzugsweise nicht zu unterbrechender Programmschrittfolgen innerhalb seiner Programmsequenz einen ihm zugeordneten Riegel setzt und nach Abarbeitung solcher Programmschrittfolgen den ihm zugeordneten Riegel wieder aufhebt,
35 - daß nach Ablauf des dem Laufzeitbegrenzungswert entsprechenden Zeitraums der untergeordnete Rechenprozeß zur

- Abarbeitung anderer anstehender nicht zeitkritischer Rechenprozesse unterbrochen wird, sofern kein dem untergeordneten Rechenprozeß zugeordneter Riegel gesetzt ist,
- daß nach Ablauf eines von dem Laufzeitbegrenzungswert ableitbaren größeren Zeitraums, der untergeordnete Rechenprozeß zur Abarbeitung anderer anstehender nicht zeitkritischer Rechenprozesse unterbrochen wird, unabhängig davon, ob ein dem untergeordneten Rechenprozeß zugeordneter Riegel gesetzt ist oder nicht,
 - daß bei Aufhebung, des ihm zugeordneten Riegels in dem Zeitintervall zwischen dem durch den Laufzeitbegrenzungswert vorgegebenen Zeitraum und dem vom Laufzeitbegrenzungswert ableitbaren größeren Zeitraum, der untergeordnete Rechenprozeß sofort zur Abarbeitung anderer anstehender nicht zeitkritischer Rechenprozesse unterbrochen wird,
 - daß der untergeordnete Rechenprozeß, wie jeder andere nicht zeitkritische Rechenprozeß, von einem anstehende zeitkritischen Rechenprozeß immer sofort unterbrochen wird.
- Durch die Laufzeitbegrenzung einer nicht zeitkritischen Programmsequenz wird sichergestellt, daß das Echtzeitsystem spätestens nach Ablauf eines vordefinierbaren Zeitraums diese Programmsequenz unterbrechen kann, um andere nicht zeitkritische Rechenprozesse weiter abarbeiten zu können. Dadurch wird sichergestellt, daß eine rechenzeitaufwendige Programmsequenz andere nicht zeitkritische Rechenprozesse nicht länger als einen vordefinierbaren Zeitraum lang aufhalten kann.
- Die erfindungsgemäße Umsetzung der Laufzeitbegrenzung durch einen untergeordneten Rechenprozeß mit Riegelsetzung braucht keine aufwendige Verwaltung, ist auch für Programmsequenzen geeignet, die nicht immer im Voraus lokalisierbar sind und kann prinzipiell auch während der Laufzeit vorgenommen werden ohne die Systemkomplexität zu erhöhen.

Eine vorteilhafte Weiterbildung der Erfindung besteht darin, daß der untergeordnete Rechenprozeß eine Rückmeldung an den ursprünglichen Rechenprozeß übergibt, ob er sich nach Abarbeitung selbst beendet hat, oder ob er aufgrund seiner Laufzeitbegrenzung unterbrochen wurde. Das hat den Vorteil, daß beispielsweise mit der nächsten Aktivierung die Laufzeitbegrenzung wieder neu festgelegt werden kann. Eine Optimierung der Rechenzeitverteilung ist daher auch zur Laufzeit noch leicht vornehmbar.

10

Ein Ausführungsbeispiel der Erfindung ist in den Zeichnungen dargestellt und wird im folgenden näher erläutert. Dabei zeigen:

15 FIG 1 eine Blockdarstellung einer auf einem Echtzeitsystem basierenden numerischen Steuerung im Zusammenwirken mit einer Werkzeugmaschine,

FIG 2 die Rechenzeiteinteilung zwischen zeitkritischen Rechenprozessen und nicht zeitkritischen Rechenprozessen,

20 FIG 3a eine symbolische Darstellung von vier nicht zeitkritischen Rechenprozessen vor Anwendung des erfindungsgemäßen Verfahrens,

FIG 3b eine symbolische Darstellung der nicht zeitkritischen Rechenprozesse bei Anwendung des erfindungsgemäßen Verfahrens und

25

FIG 4 in diesem Zusammenhang eine symbolische Darstellung eines untergeordneter Rechenprozeßes bei unterschiedlichen Laufzeitbegrenzungswerten.

30 Die Darstellung gemäß FIG 1 zeigt eine Blockdarstellung einer auf einem Echtzeitsystem NCK basierenden numerischen Steuerung. Das Echtzeitsystem NCK ist mit einer Werkzeugmaschine WM, mit einer Ein- und Ausgabeeinheit PC und einem Informationsbus BU verbunden. Im Echtzeitsystem NCK sind mehrere Rechenprozesse als Rechtecke Rz1, Rz2, R10, R11, R12, R13 abgebildet. Dabei sind die Rechenprozesse Rz1 und Rz2 für die regel-

35

mässige und zeitkritische Ausgabe von Sollwerten an die Antriebe der Werkzeugmaschine WM und der ebenfalls regelmäßigen und zeitkritischen Erfassung der Lage- und Drehzahlgeberwerte zuständig. Andere Rechenprozesse sind zuständig für das Aktualisieren der Anzeigeeinheit (Rechenprozeß R13) das Abfragen der Tastatur der Eingabeeinheit (Rechenprozeß R11) das Einlesen von Teileprogrammen (Rechenprozeß R12) und die Aufbereitung von NC-Daten aufgrund der Teileprogramme (Rechenprozeß R10) und sind dementsprechend nicht zeitkritisch.

10

Die Darstellung gemäß FIG 2 zeigt ein waagerechtes Balkendiagramm, bei dem die Zeit entlang der waagerechten Achse in Bezugspfeilrichtung zunimmt. Die vorhandene Rechenzeit der Steuerung ist in einzelne Abschnitte unterteilt, in denen immer nur ein einziger Rechenprozeß auf einmal abgearbeitet wird. Die mit Rz1 gekennzeichneten Abschnitte entsprechen einem zyklisch wiederkehrenden zeitkritischen Rechenprozeß, der beispielsweise für die Lageregelung einer Achse an der Werkzeugmaschine zuständig ist. Die mit Rz2 gekennzeichneten Abschnitte entsprechen einem zweiten zyklisch wiederkehrenden zeitkritischen Rechenprozeß, der beispielsweise für die Drehzahlregelung einer weiteren Achse an der Werkzeugmaschine zuständig ist. Zur Vereinfachung der Darstellung werden nur zwei zeitkritische Rechenprozesse Rz1 und Rz2 gezeigt. In den Zeiträumen FZI zwischen der Abarbeitung der zeitkritischen Rechenprozesse muß die Abarbeitung aller nicht zeitkritischen Rechenprozesse untergebracht werden.

In FIG 3a sind beispielsweise vier nicht zeitkritische Rechenprozesse R10, R11, R12, R13 symbolisch als Balken dargestellt. Die Länge der Balken entspricht der relativen Rechenzeit, die zur vollständigen Abarbeitung des jeweiligen Prozesses benötigt wird. So benötigt der Rechenprozeß R13 am wenigsten Rechenzeit und der Rechenprozeß R12 mit Abstand am meisten.

35

Das Echtzeitsystem wählt nach dem Fachmann bekannten Verfahren einen der nicht zeitkritischen Rechenprozesse R10, R11, R12, R13 aus und arbeitet diesen in den noch freien Zeiträumen FZI zwischen der Abarbeitung der zyklisch wiederkehrenden zeitkritischen Rechenprozesse Rz1 und Rz2 ab.

Jeder Rechenprozeß besteht aus mindestens einer, aus abzuarbeitenden Programmschrittfolgen bestehenden Programmsequenz. So besteht z.B. der Rechenprozeß R12 aus den drei Programmsequenzen PS1, PS2 und PS3, wobei beispielsweise die Programmsequenz PS2 lastabhängig im Vergleich zu den anderen Programmsequenzen PS1 und PS3 unverhältnismäßig groß sein möge, bzw. unverhältnismäßig viel Rechenzeit benötigen möge. Um die Darstellung zu vereinfachen, sind in FIG 3a die Programmsequenzen der anderen Rechenprozesse nicht gezeigt.

Wenn das Echtzeitsystem den Rechenprozeß R12 wählt, um ihn in den Zeiträumen FZI abzuarbeiten, so entsteht das Problem, daß eine einzelne Programmsequenz wie PS2 die gesamte noch zur Verfügung stehenden Rechenzeit FZI auf geraume Zeit aufbraucht und alle anderen Programmsequenzen der nicht zeitkritischen Rechenprozesse R10, R11, R13 vorerst stehen bleiben.

Um dieses zu verhindern, wird erfindungsgemäß eine nicht zeitkritische Programmsequenz wie PS2 mit einer Laufzeitbegrenzung versehen. Das Echtzeitsystem kann dann die Programmsequenz PS2 unterbrechen, um einen der anderen nicht zeitkritischen Rechenprozesse R10, R11, R13 weiter abzuarbeiten. Wie in FIG 3b gezeigt, wird dazu die Programmsequenz PS2 als ein dem Rechenprozeß R12 untergeordneter Rechenprozeß R14 abgesetzt (siehe Pfeil P) und im Echtzeitsystem mit einem Laufzeitbegrenzungswert LZG versehen.

In FIG 4 wird nun symbolisch dargestellt, wie der untergeordnete Rechenprozeß R14 abgearbeitet wird. Die Zeit Z ist in FIG 4 als waagerechte Achse mit Zeitpunkten t0, t1, etc. bis

t15 abgebildet. Darunter symbolisiert der Balken R14 von t0 bis t14 den laufzeitbegrenzten untergeordneten Rechenprozeß R14.

- 5 Im Inneren des Balken R14 sind die Zeitabschnitte t1 bis t2, t3 bis t9 und t12 bis t13 mit R gekennzeichnet und stellen Programmschrittfolgen innerhalb der Programmsequenz PS2 dar, die vorzugsweise nicht unterbrochen werden sollten. Bei einem Rechenprozeß der Daten in Paketen überträgt, entspräche bei-
- 10 spielsweise die Übertragung eines Datenpaketes einer solchen nicht zu unterbrechenden Programmschrittfolge. Wird mitten in der Übertragung eines Datenpakets der Prozeß unterbrochen, so muß das ganze Paket zu einem späteren Zeitpunkt nochmals übertragen werden, was wenig effizient ist. Wird dagegen erst
- 15 nach Abschluß eines Pakets unterbrochen, so kann zum späteren Zeitpunkt ohne Effizienzverlust mit der Übertragung des nächsten Datenpakets fortgefahren werden.

- Aus diesem Grund wird nach dem erfindungsgemäßen Verfahren
- 20 vor Beginn vorzugsweise nicht zu unterbrechender Programmschrittfolgen innerhalb der Programmsequenz PS2 ein dem Rechenprozeß R14 zugeordneten Riegel (auch 'Lock' genannt) R, gesetzt (im Ausführungsbeispiel zu den Zeitpunkten t1, t3 und t12) und nach Abarbeitung solcher Programmschrittfolgen wird
- 25 der Riegel R wieder aufgehoben (im Ausführungsbeispiel zu den Zeitpunkten t2, t9 und t13). In der FIG 4 entsprechen daher die mit R gekennzeichneten Abschnitte im Inneren des Balken R14 den Zeitabschnitten t1 bis t2, t3 bis t9 und t12 bis t13, wo der Riegel für den Rechenprozeß R14 gesetzt wurde.

- 30 Wie der laufzeitbegrenzte Rechenprozeß R14 vom Echtzeitsystem nach Ablauf seiner Laufzeitbegrenzung LZG unterbrochen wird, hängt nun davon ab, ob zu diesem Zeitpunkt der Riegel R gesetzt ist oder nicht. Hierbei sind mehrere Fälle zu unter-
- 35 scheiden.

Im einfachsten Fall kann der Rechenprozeß R14 innerhalb seiner Laufzeitbegrenzung LZG vollkommen abgearbeitet werden und unterscheidet sich daher nicht weiter von den anderen Rechenprozessen. Aus diesem Grund wird auf diesen trivialen Fall
5 nicht weiter eingegangen.

Alle anderen Fälle sind in FIG 4 jeweils in den Abschnitten A, B und C dargestellt.

10 Im Abschnitt A der FIG 4 ist die Laufzeitbegrenzung LZG als waagerechter Doppelpfeil unterhalb vom Balken R14 für den Rechenprozeß R14 abgebildet. In diesem Fall endet die Laufzeitbegrenzung LZG am Zeitpunkt t11, zu dem der Riegel R sich im nicht gesetzten Zustand befindet. Daraufhin wird der Rechen-
15 prozeß R14 zum Zeitpunkt t11 unterbrochen, was durch das Kreuz U1 gekennzeichnet ist.

Im Abschnitt B der FIG 4 ist die Laufzeitbegrenzung LZG als waagerechter Doppelpfeil unterhalb vom Balken R14 für den Re-
20 chenprozeß R14 abgebildet. Daneben befindet sich ein weiterer strichpunktierter Doppelpfeil LZG' gleicher Länge, der eine Verdoppelung der ursprünglich vorgegebenen Laufzeitbegrenzung LZG darstellt. In diesem Fall endet die ursprüngliche Laufzeitbegrenzung LZG am Zeitpunkt t7, zu dem der Riegel R sich
25 im gesetzten Zustand befindet. Daraufhin wird der Rechenprozeß R14 zum Zeitpunkt t7 nicht unterbrochen. Statt dessen wird ihm eine Verlängerung der Laufzeitbegrenzung zugestanden, die z.B. aus einer Verdoppelung der ursprünglichen vorgegebenen Laufzeitbegrenzung von LZG auf LZG + LZG' besteht.
30 Dieser von der Laufzeitbegrenzung LZG auch auf andere Art ableitbare größere Laufzeitbegrenzungszeitraum LZG + LZG' würde erst zum Zeitpunkt t14 zu Ende sein. Die vorzugsweise nicht zu unterbrechende Programmschrittfolge ist aber zum früheren Zeitpunkt t9 bereits zu Ende. Dementsprechend hebt der Re-
35 chenprozeß R14 den Riegel R zum Zeitpunkt t9 auf. Daraufhin unterbricht das Echtzeitsystem den gegenüber seiner ursprüng-

lichen Laufzeitbegrenzung LZG bereits überzogenen Rechenprozeß R14 sofort zum Zeitpunkt t9, was durch das Kreuz U2 gekennzeichnet ist.

- 5 Im Abschnitt C der FIG 4 ist die Laufzeitbegrenzung LZG als waagerechter Doppelpfeil unterhalb vom Balken R14 für den Rechenprozeß R14 abgebildet. Daneben befindet sich ein weiterer strichpunktierter Doppelpfeil LZG' gleicher Länge, der eine Verdoppelung der ursprünglich vorgegebenen Laufzeitbegrenzung
- 10 LZG darstellt. In diesem Fall endet die ursprüngliche Laufzeitbegrenzung LZG am Zeitpunkt t4 zu dem der Riegel R sich im gesetzten Zustand befindet. Daraufhin wird der Rechenprozeß R14 zum Zeitpunkt t4 nicht unterbrochen. Statt dessen wird ihm eine Verlängerung der Laufzeitbegrenzung zugestanden,
- 15 die z.B. aus einer Verdoppelung der ursprünglichen vorgegebenen Laufzeitbegrenzung von LZG auf $LZG + LZG'$ besteht. Dieser von der Laufzeitbegrenzung LZG auch auf andere Art ableitbare größere Laufzeitbegrenzungszeitraum $LZG + LZG'$ ist erst zum Zeitpunkt t8 abgeschlossen. Die vorzugsweise nicht
- 20 zu unterbrechende Programmschrittfolge ist aber zu diesem Zeitpunkt noch nicht zu Ende. Dennoch unterbricht das Echtzeitsystem den gegenüber seiner ursprünglichen Laufzeitbegrenzung LZG bereits überzogenen Rechenprozeß R14 zum Zeitpunkt t8, was durch das Kreuz U3 gekennzeichnet ist.

25

In allen Fällen kann dabei eine entsprechende Rückmeldung dem übergeordneten Rechenprozeß R12 übergeben werden, damit dieser bei der nächsten Aktivierung eine eventuelle Optimierung der Laufzeitbegrenzung LZG vornehmen kann.

Patentansprüche

1. Verfahren zum Betrieb einer ein Echtzeitsystem aufweisen-
den numerischen Steuerung, insbesondere für Werkzeugmaschinen
5 oder Roboter, wobei das Echtzeitsystem zeitkritische Rechen-
prozesse (Rz1,Rz2) und nicht zeitkritische Rechenprozesse
(R10,R11,R12, R13) mit je mindestens einer, aus abzuarbeitenden
Programmschrittfolgen bestehenden, Programmsequenz (PS1,
PS2,PS3) aufweist, d a d u r c h g e k e n n -
10 z e i c h n e t,
- daß zur Laufzeitbegrenzung einer Programmsequenz (PS2)
eines nicht zeitkritischen Rechenprozesses (R12), die
Programmsequenz (PS2) als ein dem Rechenprozeß (R12)
untergeordneter Rechenprozeß (R14) mit einem Laufzeitbe-
15 grenzungswert (LZG) abgesetzt und so abgearbeitet wird,
 - daß der untergeordnete Rechenprozeß (R14) vor Beginn vor-
zugsweise nicht zu unterbrechender Programmschrittfolgen
innerhalb seiner Programmsequenz (PS2) einen ihm zugeord-
neten Riegel (R) setzt und nach Abarbeitung solcher Pro-
20 grammschrittfolgen den ihm zugeordneten Riegel (R) wieder
aufhebt,
 - daß nach Ablauf des dem Laufzeitbegrenzungswert (LZG) ent-
sprechenden Zeitraums der untergeordnete Rechenprozeß (R14)
zur Abarbeitung anderer anstehender nicht zeitkritischer
25 Rechenprozesse (R10,R11,R13) unterbrochen wird, sofern kein
dem untergeordneten Rechenprozeß zugeordneter Riegel (R)
gesetzt ist,
 - daß nach Ablauf eines von dem Laufzeitbegrenzungswert (LZG)
ableitbaren größeren Zeitraums, der untergeordnete Rechen-
30 prozeß (R14) zur Abarbeitung anderer anstehender nicht
zeitkritischer Rechenprozesse (R10,R11,R13) unterbrochen
wird, unabhängig davon, ob ein dem untergeordneten Rechen-
prozeß (R14) zugeordneter Riegel (R) gesetzt ist oder
nicht,

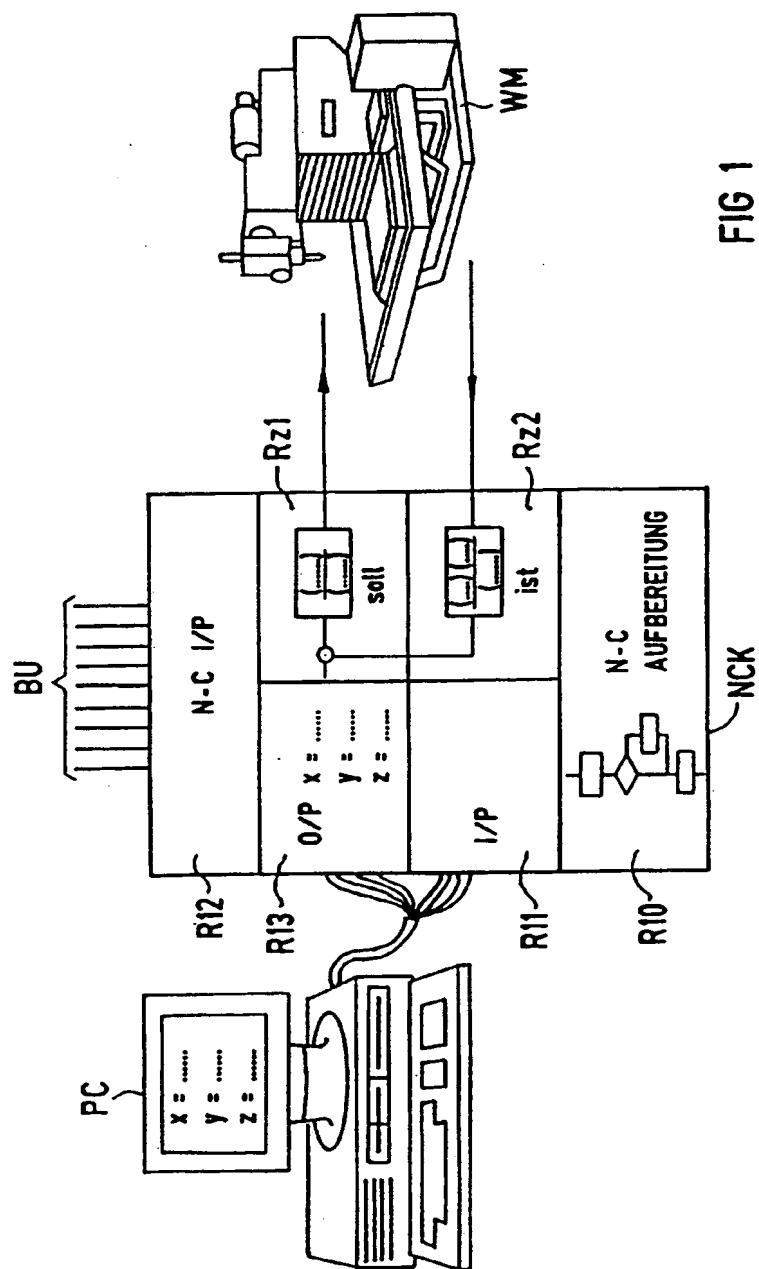
12

- daß bei Aufhebung, des ihm zugeordneten Riegels (R) in dem Zeitintervall zwischen dem durch den Laufzeitbegrenzungswert (LZG) vorgegebenen Zeitraum und dem vom Laufzeitbegrenzungswert ableitbaren größeren Zeitraum der untergeordnete Rechenprozeß (R14) sofort zur Abarbeitung anderer anstehender nicht zeitkritischer Rechenprozesse (R10,R11, R13) unterbrochen wird,
- daß der untergeordnete Rechenprozeß (R14), wie jeder andere nicht zeitkritische Rechenprozeß (R10,R11,R12,R13) von einem anstehenden zeitkritischen Rechenprozeß (Rz1,Rz2) immer sofort unterbrochen wird.

2. Verfahren nach Anspruch 1 d a d u r c h g e k e n n -
z e i c h n e t, daß der untergeordnete Rechenprozeß (R14)
eine Rückmeldung an den ursprünglichen Rechenprozeß (R12)
übergibt, ob er sich nach Abarbeitung selbst beendet hat,
oder ob er aufgrund seiner Laufzeitbegrenzung (LZG) unter-
brochen wurde.

20

1/3



2/3

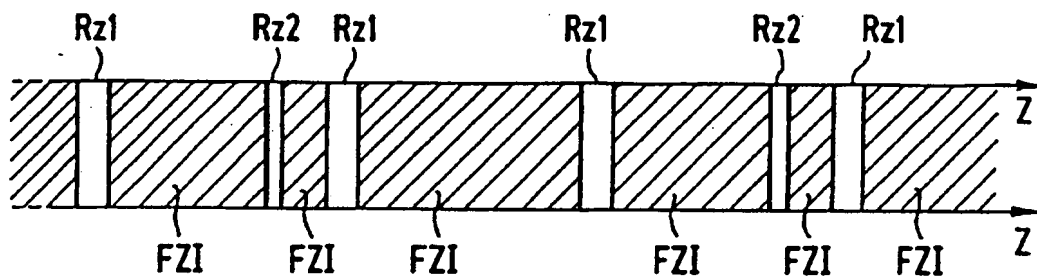


FIG 2

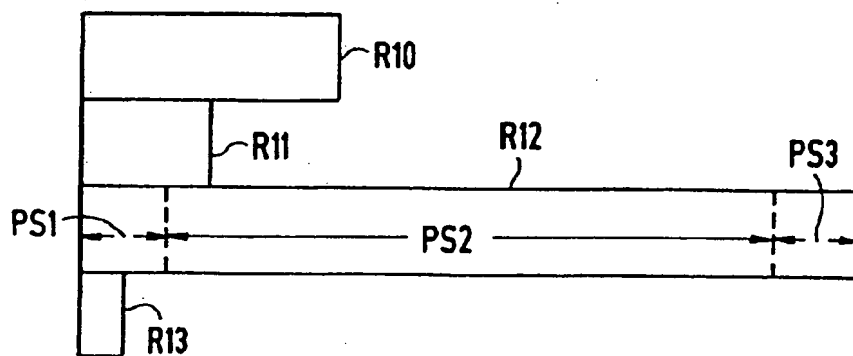


FIG 3a

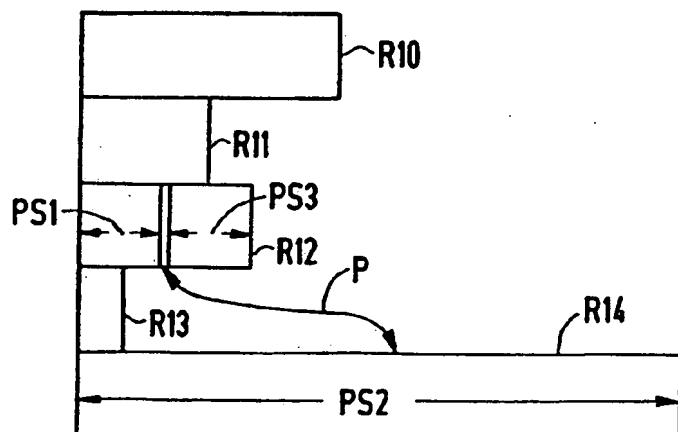


FIG 3b

3/3

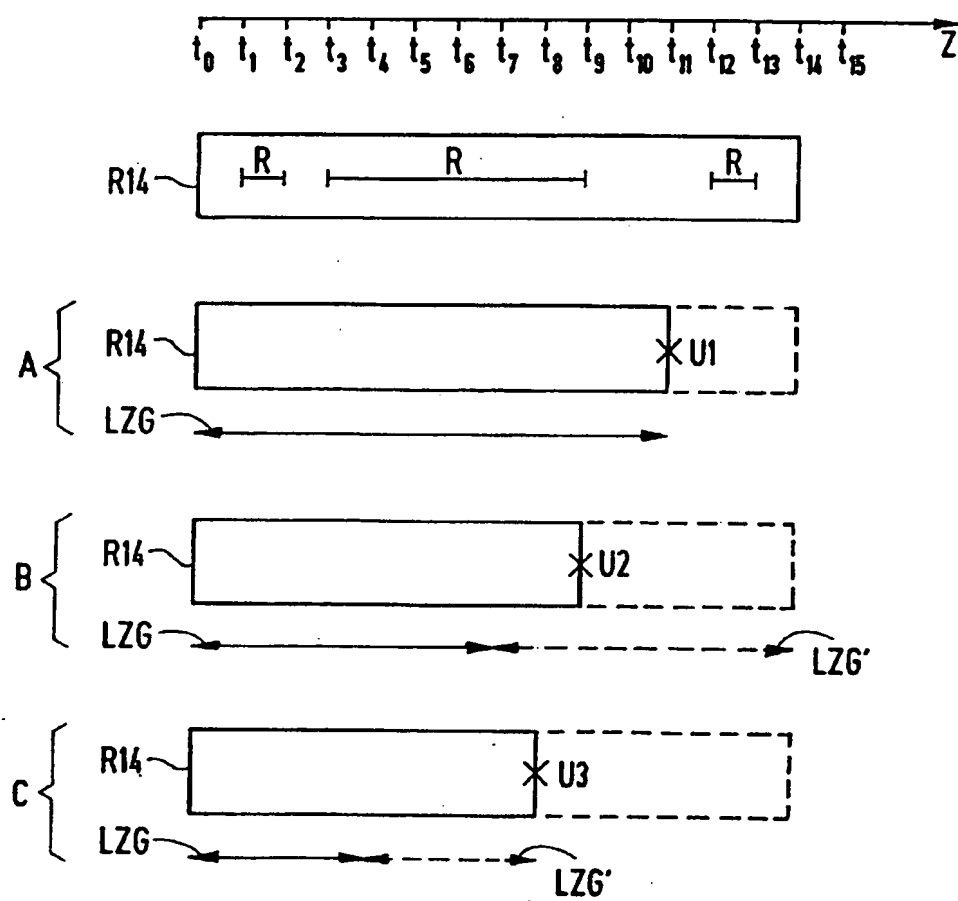


FIG 4

INTERNATIONAL SEARCH REPORT

International Application No

PC/DE 96/00619

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 G05B19/05

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 6 G05B

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP,A,0 636 956 (SIEMENS AG) 1 February 1995 see column 2, line 34 - column 3, line 31 ---	1,2
A	US,A,4 638 452 (SCHULTZ RONALD E ET AL) 20 January 1987 see column 2, line 27 - line 34 see column 10, line 55 - column 11, line 20 -----	1,2

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

* Special categories of cited documents:

- *A* document defining the general state of the art which is not considered to be of particular relevance
- *E* earlier document but published on or after the international filing date
- *L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- *O* document referring to an oral disclosure, use, exhibition or other means
- *P* document published prior to the international filing date but later than the priority date claimed

- *T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- *X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- *Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- *A* document member of the same patent family

Date of the actual completion of the international search

8 August 1996

Date of mailing of the international search report

3 0. 08. 96

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax (+31-70) 340-3016

Authorized officer

Messelken, M

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/DE 96/00619

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP-A-0636956	01-02-95	NONE	
US-A-4638452	20-01-87	NONE	

INTERNATIONALER RECHERCHENBERICHT

Internationales Aktenzeichen
PCI/DE 96/00619

A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
IPK 6 G05B19/05

Nach der internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

B. RECHERCHIERTE GEBIETE

Recherchierte Mindestprüfstoff (Klassifikationssystem und Klassifikationsymbole)
IPK 6 G05B

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A	EP,A,0 636 956 (SIEMENS AG) 1.Februar 1995 siehe Spalte 2, Zeile 34 - Spalte 3, Zeile 31 ---	1,2
A	US,A,4 638 452 (SCHULTZ RONALD E ET AL) 20.Januar 1987 siehe Spalte 2, Zeile 27 - Zeile 34 siehe Spalte 10, Zeile 55 - Spalte 11, Zeile 20 -----	1,2

☐ Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen

☒ Siehe Anhang Patentfamilie

* Besondere Kategorien von angegebenen Veröffentlichungen :

"A" Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist

"E" älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist

"L" Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)

"O" Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht

"P" Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

"T" Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist

"X" Veröffentlichung von besonderer Bedeutung, die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden

"Y" Veröffentlichung von besonderer Bedeutung, die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann nahelegend ist

"A" Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche

8.August 1996

Abenddatum des internationalen Recherchenberichts

30.08.96

Name und Postanschrift der Internationale Recherchenbehörde
Europäisches Patentamt, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+ 31-70) 340-2040, Tx. 31 651 epo nl,
Fax (+ 31-70) 340-3016

Bevollmächtigter Beauftragter

Messelken, M

INTERNATIONALER RECHERCHENBERICHT

Angaben zu Veröffentlichungen, die zur selben Patentfamilie gehören

Internationales Aktenzeichen

PCT/DE 96/00619

Im Recherchenbericht angeführtes Patentdokument	Datum der Veröffentlichung	Mitglied(er) der Patentfamilie	Datum der Veröffentlichung
EP-A-0636956	01-02-95	KEINE	
US-A-4638452	20-01-87	KEINE	

1/9/1

DIALOG(R)File 351:Derwent WPI

(c) 2007 The Thomson Corporation. All rts. reserv.

0007855657 - Drawing available

WPI ACC NO: 1996-485913/199648

XRPX Acc No: N1996-409302

Real-time numerical control operation with time-critical computations
- is

based on limited running time of non-time-critical program sequence
subordinated for liability to interruption unless protected by lock.

Patent Assignee: SIEMENS AG (SIEI)

Inventor: REHM T; WILLERT J

Patent Family (5 patents, 21 countries)

Patent Number	Kind	Date	Application Number	Kind	Date	Update
WO 1996033450	A1	19961024	WO 1996DE619	A	19960409	199648 B
EP 824726	A1	19980225	EP 1996908023	A	19960409	199812 E
			WO 1996DE619	A	19960409	
US 5909371	A	19990601	WO 1996DE619	A	19960409	199929 E
			US 1998945489	A	19980202	
KR 1999007812	A	19990125	WO 1996DE619	A	19960409	200014 E
			KR 1997707335	A	19971017	
CN 1181139	A	19980506	CN 1996193152	A	19960409	200236 E

Priority Applications (no., kind, date): DE 19514471 A 19950419

Patent Details

Number	Kind	Lan	Pg	Dwg	Filing Notes
WO 1996033450	A1	DE	18	4	
National Designated States,Original: CN KR US					
Regional Designated States,Original: AT BE CH DE DK ES FI FR GB GR IE IT					
LU MC NL PT SE					
EP 824726	A1	DE			PCT Application WO 1996DE619
					Based on OPI patent WO 1996033450
Regional Designated States,Original: AT CH DE ES FR GB IT LI SE					
US 5909371	A	EN			PCT Application WO 1996DE619
					Based on OPI patent WO 1996033450
KR 1999007812	A	KO	5		PCT Application WO 1996DE619
					Based on OPI patent WO 1996033450

Alerting Abstract WO A1

The method involves limitation of the running time of a program sequence of a non-time-critical process (R12) by treatment of the sequence as a subordinate process (R14) with a time limit (LZG). On expiry of the latter, the subordinate process is interrupted in favour of other non-time-critical processes (R10,R11,R13) as long as no lock (R) on the subordinate process is applied. The initial limit of running time can be overridden only within predetermined limits.

USE - Particularly for machine tools or robots, strongly load-dependent

program sequences which are not time-critical are prevented from taking up any time between time-critical computations.

Title Terms /Index Terms/Additional Words: REAL-TIME; NUMERIC; CONTROL;
OPERATE; TIME; CRITICAL; COMPUTATION; BASED; LIMIT; RUN; NON; PROGRAM;
SEQUENCE; LIABLE; INTERRUPT; PROTECT; LOCK

Class Codes

International Classification (Main): G05B-019/05, G06F-019/00
US Classification, Issued: 364167020, 364474110, 395670000, 395737000

File Segment: EPI;
DWPI Class: T06; X25
Manual Codes (EPI/S-X): T06-A04A; T06-D06; T06-D07B; X25-A03E; X25-A03F

Original Publication Data by Authority

China

Publication No. CN 1181139 A (Update 200236 E)
Publication Date: 19980506
Assignee: SIEMENS AG; DE (SIEI)
Language: ZH
Application: CN 1996193152 A 19960409 (Local application)
Priority: DE 19514471 A 19950419

EPO

Publication No. EP 824726 A1 (Update 199812 E)
Publication Date: 19980225
**VERFAHREN ZUM BETRIEB EINER NUMERISCHEN STEUERUNG MIT ZEITKRITISCHEN UND

NICHT ZEITKRITISCHEN PROZESSEN IN EINEM ECHTZEITSYSTEM
PROCESS FOR OPERATING A NUMERICAL CONTROL WITH CRITICAL-TIME AND
NON-CRITICAL-TIME PROCESSES IN A REAL TIME SYSTEM
PROCEDE PERMETTANT DE FAIRE FONCTIONNER UNE COMMANDE NUMERIQUE AVEC
DES

PROCESSUS A DUREE CRITIQUE OU NON, DANS UN SYSTEME EN TEMPS REEL**
Assignee: SIEMENS AKTIENGESSELLSCHAFT, Wittelsbacherplatz 2, 80333
Muenchen,
DE (SIEI)

Inventor: REHM, Thomas, Am Erlanger Weg 45, D-91052 Erlangen, DE
WILLERT, Josef, Auracher Strasse 1, D-91093 He dorf, DE
Language: DE

Application: EP 1996908023 A 19960409 (Local application)
WO 1996DE619 A 19960409 (PCT Application)
Priority: DE 19514471 A 19950419

Related Publication: WO 1996033450 A (Based on OPI patent)
Designated States: (Regional Original) AT CH DE ES FR GB IT LI SE
Original IPC: G05B-19/05(A)
Current IPC: G05B-19/05(A)

Original Abstract: In order, in a real-time system of a digital control,

to prevent an individual program sequence (PS2) of a non-critical computing process ("task") from using up a load-dependently

disproportionate amount of the total computing time still available,
that program sequence (PS2) is classed as a subordinate computing process (R14) with a running time limit (LZF). Program sequences which should preferably not be interrupted are protected against interruption by a lock. The original running time limit (LZG) can be overcome only within predetermined limits.

Korea

Publication No. KR 1999007812 A (Update 200014 E)
Publication Date: 19990125
Assignee: SIEMENS AG (SIEI)
Language: KO (5 drawings)
Application: WO 1996DE619 A 19960409 (PCT Application)
KR 1997707335 A 19971017 (Local application)
Priority: DE 19514471 A 19950419
Related Publication: WO 1996033450 A (Based on OPI patent)
Original IPC: G05B-19/05 (A)
Current IPC: G05B-19/05 (A)

United States

Publication No. US 5909371 A (Update 199929 E)
Publication Date: 19990601
Process of operating a numerical control with critical-time and non-critical-time processes in a real time system.
Assignee: Siemens Aktiengesellschaft, Munich, DE (SIEI)
Inventor: Willert, Josef, Hessdorf, DE
Rehm, Thomas, Erlangen, DE
Agent: Kenyon Kenyon
Language: EN
Application: WO 1996DE619 A 19960409 (PCT Application)
US 1998945489 A 19980202 (Local application)
Priority: DE 19514471 A 19950419
Related Publication: WO 1996033450 A (Based on OPI patent)
Original IPC: G06F-19/00 (A)
Current IPC: G06F-19/00 (A)
Original US Class (main): 364167.02
Original US Class (secondary): 364474.11 395670 395737
Original Abstract: In a real-time system of a digital control system, in order to prevent a single program sequence (PS2) of a non-time-critical computing process ("task") from using up a load-dependently disproportionate amount of the total computing time still available, the program sequence (PS2) is classified as a subordinate computing process (R14) that has a running-time limit (LZF). Sequences of program steps that are preferably not to be interrupted are in this case protected from interruption by a lock, the original running-time limit (LZG) being able to be run over only within predetermined limits.
Claim:

1.A method for operating a numerical control system having a real-time system, the real-time system having time-critical computing processes and non-time-critical computing processes, each of the computing processes having at least one program sequence, the method comprising the steps of:

- * (a) classifying a program sequence of a non-time-critical computing process as a subordinate computing process to limit a running time of the program sequence, the subordinate computing process being subordinate to the respective non-time-critical computing process and having a running-time limit value associated therewith;
- * (b) setting a lock indicator assigned to the subordinate computing process before starting a sequence of program steps;
- * (c) after processing the sequence of program steps, canceling the lock indicator assigned to the subordinate computing process;
- * (d) after a time period corresponding to the running-time limit value elapses, interrupting the subordinate computing process to process at least another non-time-critical computing process that is pending if the lock indicator assigned to the subordinate computing process is not set;
- * (e) after a further time period greater than the running-time limit value elapses, interrupting the subordinate computing process to process another of the non-time-critical computing processes that is pending;
- * (f) interrupting the subordinate computing process to process another of the non-time-critical computing processes that is pending if the lock indicator assigned to the subordinate computing process is reset in a time interval between the running-time limit value and the further time period; and
- * (g) interrupting any non-time-critical computing process if a time-critical computing process is pending.

WIPO

Publication No. WO 1996033450 A1 (Update 199648 B)
Publication Date: 19961024

****PROCESS FOR OPERATING A NUMERICAL CONTROL WITH CRITICAL-TIME AND
NON-CRITICAL-TIME PROCESSES IN A REAL TIME SYSTEM****

Assignee: SIEMENS AKTIENGESSELLSCHAFT, DE (SIEI)

Inventor: REHM, THOMAS, DE

WILLERT, JOSEF, DE

Language: DE (18 pages, 4 drawings)

Application: WO 1996DE619 A 19960409 (Local application)

Priority: DE 19514471 A 19950419

Designated States: (National Original) CN KR US

(Regional Original) AT BE CH DE DK ES FI FR GB GR IE IT LU MC NL PT

SE

Original IPC: G05B-19/05(A)

Current IPC: G05B-19/05(A)

Original Abstract:

Um bei einem Echtzeitsystem einer digitalen Steuerung zu verhindern,
dass

eine einzelne Programmsequenz (PS2) eines nicht zeitkritischen
Rechenprozesses ("Task") lastabhaengig unverhaeltnismaessigerweise

die

gesamte noch zur Verfuegung stehende Rechenzeit aufbraucht, wird

diese

Programmsequenz (PS2) als ein untergeordneter Rechenprozess (R14)

mit

einer Laufzeitbegrenzung (LZF) abgesetzt. Vorzugsweise nicht zu
unterbrechende Programmschrittfolgen werden dabei ueber einen

Riegel

("Lock") vor Unterbrechung geschuetzt, wobei die urspruengliche
Laufzeitbegrenzung (LZG) nur innerhalb vorbestimmter Grenzen

ueberzogen

werden kann. In order, in a real-time system of a digital control,

to

prevent an individual program sequence (PS2) of a non-critical
computing process ("task") from using up a load-dependently
disproportionate amount of the total computing time still

available,

that program sequence (PS2) is classed as a subordinate computing
process (R14) with a running time limit (LZF). Program sequences

which

should preferably not be interrupted are protected against

interruption

by a lock. The original running time limit (LZG) can be overcome

only

within predetermined limits.

?